An Adaptive Sampling Scheme to Efficiently Train Fully Convolutional Networks for Semantic Segmentation

Lorenz Berger[1,2], Eoin Hyde[1,2], M. Jorge Cardoso[1], and Sébastien Ourselin[1]

[1] WEISS, University College London, London, UK
[2] Innersight Labs, London, UK

Abstract

Deep convolutional neural networks (CNNs) have shown excellent performance in object recognition tasks and dense classification problems such as semantic segmentation. However, training deep neural networks on large and sparse datasets is still challenging and can require large amounts of computation and memory. In this work, we address the task of performing semantic segmentation on large data sets, such as three-dimensional medical images. We propose an adaptive sampling scheme that uses a-posterior error maps, generated throughout training, to focus sampling on difficult regions, resulting in improved learning. Our contribution is threefold: 1) We give a detailed description of the proposed sampling algorithm to speed up and improve learning performance on large images. 2) We propose a deep dual path CNN that captures information at fine and coarse scales, resulting in a network with a large field of view and high resolution outputs. 3) We show that our method is able to attain new state-of-the-art results on the VISCERAL Anatomy benchmark.

1 Introduction

This paper addresses the problem of efficiently training convolutional neural networks (CNNs) on large and imbalanced datasets. We propose a training strategy that adaptively samples the training data to effectively speed up training and avoid over-sampling data that contains little extra information. In this work, we investigate the problem of automatic segmentation from high resolution 3D CT scans. Several deep learning techniques [1–4] have recently been proposed for 3D segmentation of medical datasets. To overcome the problem of dealing with these large datasets, such as Computed Tomography (CT) volumes, commonly of dimension $512 \times 512 \times 700$, previous approaches train a CNN on a cropped region of interest which reduces the size of individual training images by around 100 fold [2, 4]. By reducing the size of training images, they can now be fit into memory and a network can be trained effectively on the selected data. However, identifying regions of interest requires an additional pre-processing step which may not be easy in many applications. Also, training CNNs on cropped images limits the field of view of the CNN and subsequently can introduce unwanted image boundary induced effects during testing. Other applications, where training CNNs on

very large images is a problem, includes the segmentation of histology datasets or the segmentation of aerial images. For example in aerial image segmentation, training a CNN to segment ships can be difficult because large portions of the image contain water which provide little information during training, resulting in slow learning. Some ideas to address this have already been proposed, for example in [3] a fixed, hand-crafted, pre-computed weight map is used to help learn small separation borders between touching cells for biomedical image segmentation. In this work the proposed sampling scheme ends up dynamically learning such a weight mapping, making it generally applicable to many applications. Curriculum learning [5] and derivative methods like self-paced learning [6] build on the intuition that, rather than considering all samples simultaneously, the algorithm should be presented with the training data in a meaningful order that facilitates learning. These ideas have already successfully been applied to image classification [7], by ordering images from easy to hard during training. Also for the problem of weakly supervised semantic segmentation [8] similar ideas are applied, where predictions from previous training iterations are used to iteratively learn segmentation maps from just a single class label per image. The focus of this paper is fully supervised semantic segmentation where a representative training set is available with dense manual label annotations and the challenge lies in efficiently learning from large datasets. We give a detailed account of the implementation, which is a straightforward extension to any existing CNN segmentation system, and present state-of-the-art segmentation results on the VISCERAL anatomy benchmark.

2 Methods

Neural Network Architecture Compared to the 3D dual path network outlined in [1], we further develop the architecture by replacing the standard convolution layers with popular resnet blocks [9], and increase the maximum network depth from 11 layers in [1] to 19 layers. By having a deeper network and a down sampled pathway with input resolution 1/4 of the original resolution, we obtain a large receptive field of size 124^3 whilst maintaining a deep high resolution pathway that does not compromise the resolution through pooling layers. The architecture (see Figure 2) results in a total of 649,251 parameters. The proposed configuration allows for a large number of samples ($3D$ patches) per batch to ensure balanced class sampling and effective optimization, whilst maintaining a deep and wide enough network to capture the high variability and spatial semantics of the data. Blocks labeled 'Conv' are standard convolutional layers with kernel size $3 \times 3 \times 3$, blocks labeled 'Res block' are standard and bottleneck resnet blocks, as detailed in [9]. Each fully connected layer is preceded by a dropout layer with probability 0.5, and a softmax non-linearity is used as a final classification layer.

Adaptive sampling strategy The problem of class imbalance as described in [1] can be dealt with by choosing small patch sizes and evenly sampling from each

class [1], and through weighted loss functions [3, 10, 11]. Both of these methods either load the whole image into GPU memory, which is not feasible for large images, or select a small subset of patches, which can lead to inefficient training on sparse datasets. To overcome both of these issues we propose the simple sampling Algorithm 1. In Algorithm 1, $\mathcal{U}(0,1)$ is a random number drawn from

Algorithm 1 isample: adaptive sampling algorithm

```
Initialize error maps for every image in the training data: E_i(x) = 1.
while CNN training do
  while training for 1 epoch do
    while filling batch with patches do
      Pick an image I_j from the training set I*.
      Pick a class k from the corresponding label map L_j.
      Pick a patch in image I_j, centered at location c, where L_j(c) = k.
      Accept patch into batch if E_i(c) > U(0,1) − ε.
    end while
    Back-propagate loss of batch and update the current CNN weights: w.
  end while

  Select a subset of images, I*, and label maps, L*, from the training set:
  for [I_k, L_k] ∈ [I*, L*] do
    Update error maps: E_k(x) = 1 − CNN(w, I_k(x))_{L_k(x)}
  end for
end while
```

the uniform distribution and $\boldsymbol{E}_i$ refers to the error map of the i^{th} training image. Error maps can easily be calculated, either after each epoch or concurrently to the training process, as

$$\boldsymbol{E}_k(\boldsymbol{x}) = 1 - \mathrm{CNN}(\boldsymbol{w}, \boldsymbol{I}_k(\boldsymbol{x}))_{\boldsymbol{L}_k(\boldsymbol{x})}, \tag{1}$$

where $\mathrm{CNN}(w, \boldsymbol{I}_k(\boldsymbol{x}))_{\boldsymbol{L}_k(\boldsymbol{x})}$ is a map of the CNN predictions over the full training image $\boldsymbol{I}_k$, evaluated using the most current weights, $\boldsymbol{w}$, and outputting the probability of the true class label $\boldsymbol{L}_k(\boldsymbol{x})$, at position $\boldsymbol{x}$. Examples of error maps produced throughout training are show in Figure 4. The additional parameter ϵ controls the strength of the isample scheme. Setting $\epsilon = 0$, corresponds to choosing patches based entirely on the amount of error that they currently produce by the network. When $\epsilon = 1$ the condition $\boldsymbol{E}_i(\boldsymbol{c}) > \mathcal{U}(0,1) - \epsilon$ is always satisfied and we are left with a standard sampling scheme that accepts every chosen patch. For all results shown in this paper we have chosen $\epsilon = 0.01$, since we are interested in using the isample scheme to full effect. Detailed investigations into how to best set this parameter for different datasets with varying amounts of sparsity is left for future work. The subset of images, $\boldsymbol{I}^*$, and label maps, $\boldsymbol{L}^*$, of the training set may be chosen in line with how quickly to introduce the isample scheme during training and the amount of computational resources available. In our experiments we had access to four GPUs, three were used to train the CNN continuously and one GPU was used in parallel to continuously perform full predictions of the validation dataset and the training dataset. From this, full

dice scores of the validation dataset and full error maps of the training dataset could be calculated. Also, having access to dice scores calculated over the full images throughout training has been helpful throughout development since these true dice scores provide more meaningful information than dice scores calculated from individual batches which are biased by the sampling scheme and the size of patches. The error maps produced can also be useful for debugging purposes.

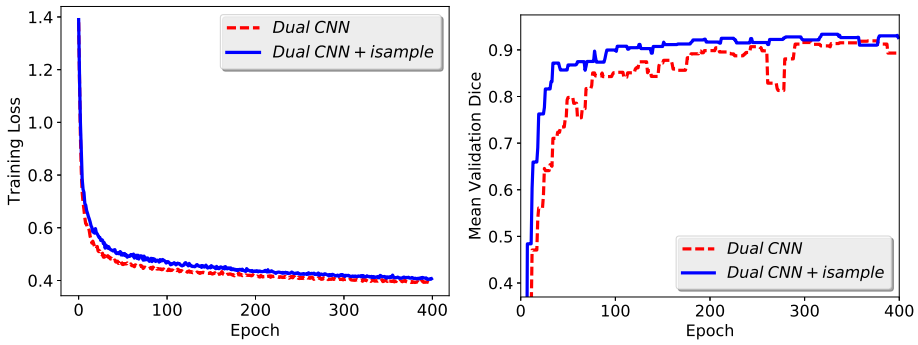
3 Results

We trained, validated and tested the automatic segmentation method on contrast enhanced CT scans from the VISCERAL Anatomy 3 dataset, made up of 20 training scans, and 10 unseen testing scans (currently not available to download) [12]. The scans are form a heterogeneous dataset with various topological changes between patients, and manual segmentations are available for a number of different anatomical structures. We randomly split the training set into 16 scans for training (80%) and 4 scans for validation (20%), we also present results of our online submission on the unseen test dataset. For illustrative purposes, the first experiment, in section 3, focuses on segmenting only the kidneys from full body CT scans. In section 3 we present results on simultaneously segmenting multiple organs from the CT data.

CNN training setup During training we perform data augmentation by resampling the 3D patches to a $[1\text{mm}, 1\text{mm}, 1.5\text{mm}] + \mathcal{U}(-0.1, 0.1)$ resolution. We also rotate each patch by $[\mathcal{U}(-10, 10), \mathcal{U}(-4, 4), \mathcal{U}(-4, 4)]$ degrees. We set voxels with values greater than 1000 to 1000, and values less than -1000 to -1000, and divide all values by a constant factor of 218 (the standard deviation of the dataset). We use Glorot initializations [13] on all convolution layers. For batch-norm layers we use the initializations technique described in [14]. We impose L_2 weight decay of size 0.0001, on all convolutional layers except on the last fully convolutional layer before the final softmax non-linearity. Using techniques described in [14] we make use of large batch sizes and large learning rates. We use SGD with Nestrov momentum set at 0.8 [15], the initial learning rate is 0.001, and each batch contains 12 patches, sampled from one randomly selected scans in the training set. We run each epoch for 100 batches. We also employ a learning rate warm up schedule as described in [14] for the first 5 epochs. We use a standard cross-entropy loss function.

Segmenting kidneys from full body CT scans In this experiment we use labels for kidneys to train the CNN, resulting in a simple two class, foreground (kidneys) and background (everything else), segmentation problem. Figure 1 shows curves of the training loss (1a) and mean validation dice score (1b) for segmented kidneys throughout training, averaged over three separate runs. The blue curve represents training runs where patches are sampled randomly but evenly from background and kidney foreground, the red curve represents training runs where patches are sampled using the proposed sampling algorithm 1.

Because isample adaptively selects more difficult patches as training progresses, the loss is higher, as seen in Figure 1a. In Figure 1b, the sampler achieves faster generalization, and our current results indicate that the final generalization of the CNN trained with the proposed sampling scheme is slightly improved for this sparse segmentation setup, where the kidneys only make up $\sim 0.3\%$ of the voxels within the whole scan. Table 1 shows the average dice scores achieved by the Dual CNN, with and without isample, throughout training. When using the isample scheme the CNN is able to achieve a dice score of 0.855 after only $5k$ training iterations. This is close to the end of training performance, a dice score of 0.899 after 40k, of the Dual CNN without isample in use.



(a) (b)

Fig. 1: (a) Training loss and (b) mean validation dice score, averaged over 3 runs.

Figure 4 shows coronal slices of a training error volume $\boldsymbol{E}_k(\boldsymbol{x})$. As seen in Figure 4c, initially there is significant error produced by the CNN prediction at epoch 16, for example misclassifying the aorta (part of the background class) because it has similar intensity values to the kidneys. After more training, at epoch 50, Figure 4d shows that the error is now much lower. The CNN has now learned that the aorta is part of the background class. However more subtle regions such as the collecting system and large vessels within the kidney (see small hole in the true segmentation of the left kidney in Figure 4b) still produce high errors, and further focused training is required to optimize the weights until they are correctly classified. There also remains a high error around the border of the kidneys, which will result in the sampling process selecting more patches from the border region, and thus ends up learning to train the network with a similar loss to the hand-crafted border weighted loss function designed in [3].

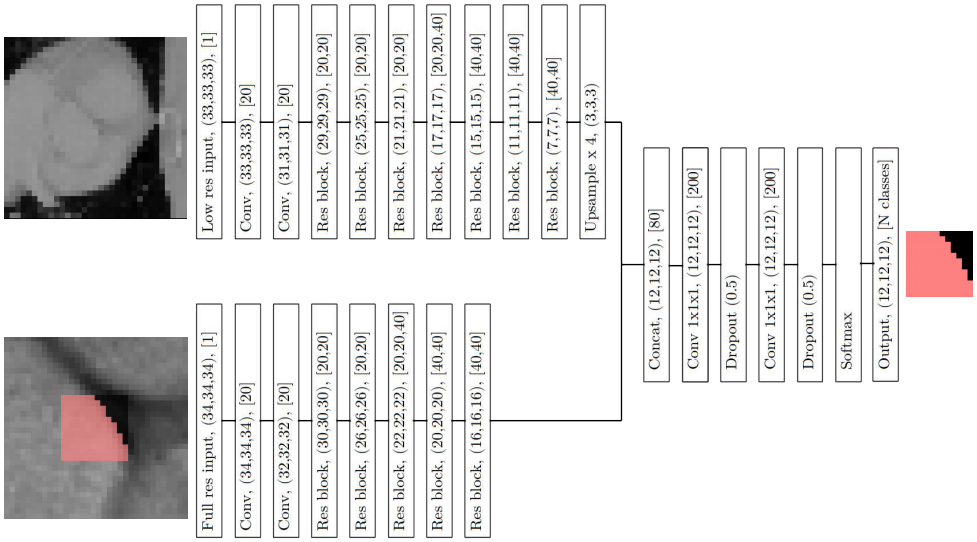


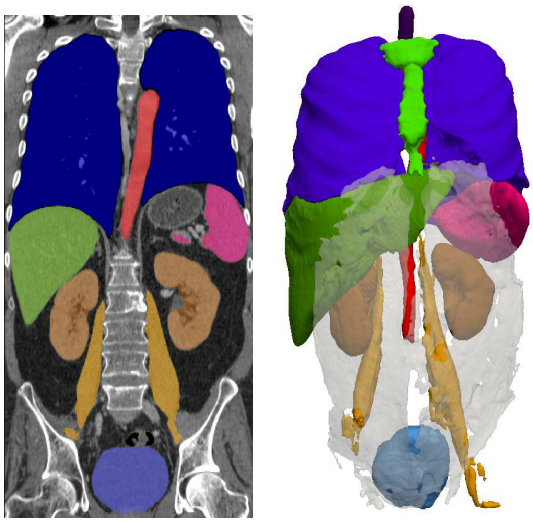
Fig. 2: The proposed dual path CNN architecture. Numbers inside round brackets give the input dimensions of each block. Numbers in square brackets refer to the number of feature maps used at each layer.

# its	Dual CNN	+ isample
5k	0.797 (0.063)	0.855 (0.048)
10k	0.849 (0.064)	0.897 (0.057)
20k	0.905 (0.054)	0.920 (0.038)
40k	0.899 (0.058)	0.927 (0.037)

Table 1: Mean dice scores and standard deviations at different number of iterations, throughout training.

Method	Kidney Dice scores
Dual CNN (validation data)	Left + Right 0.899 (0.058)
Dual CNN + isample (validation data)	Left + Right 0.927 (0.037)
Dual CNN + isample + CRF (test data)	Left 0.954 Right 0.96
Wang et al [16] (test data)	Left 0.945 (0.027) Right 0.959 (0.011)
Vincent et al [17] (test data)	Left 0.943 (0.015) Right 0.927 (0.040)
Gass et al [18] (test data)	Left 0.913 (0.029) Right 0.914 (0.027)

Table 2: Dice scores and standard deviations, where available, for different methods automatically segmenting kidneys on the VISCERAL CT enhanced dataset.



(a) (b)

Fig. 3: (a) Coronal slice of a CT scan with overlaid segmentation output, described in section 3. The organs visible in this slice are: lungs (green), liver (red), spleen (light blue), psoas major muscle (dark blue), kidneys (brown) and bladder (yellow). (b) 3D surface rendering of the segmentation.

Table 2 shows dice scores for segmenting both kidneys using different methods. The proposed method with isample performs significantly better than without. We also submitted our method, with the addition of a CRF [19] as a post-processing step, to segment the test dataset, and achieved the top score for segmenting the left and right kidneys. Inference on a full size CT scans takes $\sim$ 65 seconds using four Tesla K50 GPU cards, each with 4GB of RAM. The total training time of the model on one Tesla K50 GPU card was 5 days.

Multi-organ segmentation We now extended the previously described algorithm to include a multi-class classification output and trained the model on the main organs available on the VISCERAL CT-enhanced dataset. We post-processes the output segmentation maps (maximum class probability at each voxel), by applying a filter that only retains the largest connected binary object within the segmentation, thus removing small objects. The segmentation output of one of the validation scans is shown in Figure 3a and Figure 3b. The results of our proposed method and other state-of-the-art methods, also summarized in [12], are given in Table 3.

(a) (b) (c) (d)

Fig. 4: Coronal slices: (a) Raw CT scan from the training set. (b) Kidney segmentation overlaid onto scan. (c) Error map, $\boldsymbol{E}_k(\boldsymbol{x})$, of foreground and background classification on a training scan after 16 epochs. (d) $\boldsymbol{E}_k(\boldsymbol{x})$ after 50 epochs. For the error maps, white corresponds to voxels that are incorrectly classified and black to correctly classified voxels.

Method	Aorta	Lung	Kidney	PMajor	Liver	Abdom	Spleen	Sternum	Trachea	Bladder
Dual CNN + isample (val)	0.893	0.980	0.938	0.824	0.941	0.769	0.951	0.900	0.926	0.912
Dual CNN (val)	0.843	0.985	0.934	0.779	0.927	0.755	0.946	0.904	0.926	0.918
Ga et al [18]	0.785	0.963	0.914	0.813	0.908	-	0.781	0.635	0.847	0.683
Jimenez et al [20]	0.762	0.961	0.899	0.797	0.887	0.463	0.730	0.721	0.855	0.679
Kéchichian et al [21]	0.681	0.966	0.912	0.802	0.933	0.538	0.895	0.713	0.824	0.823
Vincent et al [17]	0.838	0.972	0.935	0.869	0.942	-	-	-	-	-
Inter-annotator agreement	0.859	0.973	0.917	0.823	0.965	0.673	0.934	0.810	0.877	0.857

Table 3: Dice scores for different automatic multi-organ segmentation methods and inter-annotator agreement results [12] on the VISCERAL dataset.

We note that because the cloud-based evaluation service [12] containing the test data was closed at the time of running these experiments, we were not able to evaluate our method on the test data, thus making direct comparisons to previous methods difficult. As previously mentioned, we trained our method on 80% of the data (16 scans) and validated it on the remaining 20% (4 scans). From having evaluated the kidney only CNN on the test data, we found that the testing dataset gave better dice scores than the validation set. We are therefore confident that our results in Table 3 are representative. The Dual CNN without using the isample scheme (average organ dice 0.8917) slightly underperformed compared to when using the isample (average organ dice 0.9034). However this difference is far less notable than during previous experiments, shown in Table 2. We hypothesize this is because the background class in the multi-organ segmentation problem is split into background and other organs such as the Lung, Liver etc, thus making the dataset, especially the background class, easier to sample from. The potential benefit of using the isample method is therefore problem dependent.

4 Conclusion

We proposed and evaluated a sampling scheme to deal with very large images such as 3D CT scans. As shown in section 3 the sampler enables fast training, and our results indicate that the final generalization performance can be improved. This is inline with previous research that shows the positive effect of curriculum learning on optimization and end performance of machine learning systems [7, 6]. Our experimental results suggests our algorithm gives new state of the art performance for the aorta, lung, kidney, rectus abdominis, spleen, sternum, trachea and bladder, on the VISCERAL anatomy benchmark, and improves upon human inter-annotator agreement scores on the following organs: aorta, lung, kidney, psoas major, rectus abdominis, spleen, sternum, trachea and bladder. These encouraging results pave the way for using CNNs for robust automatic segmentation within clinical practice, such as surgical planning.

Acknowledgments: This research was part funded by a NIHR i4i-connect grant.

References

1. K. Kamnitsas, C. Ledig, V. F. Newcombe, J. P. Simpson, A. D. Kane, D. K. Menon, D. Rueckert, and B. Glocker, "Efficient multi-scale 3d cnn with fully connected crf for accurate brain lesion segmentation," *Medical image analysis*, vol. 36, pp. 61–78, 2017.
2. P. F. Christ, M. E. A. Elshaer, F. Ettlinger, S. Tatavarty, M. Bickel, P. Bilic, M. Rempfler, M. Armbruster, F. Hofmann, M. DAnastasi, *et al.*, "Automatic liver and lesion segmentation in ct using cascaded fully convolutional neural networks and 3d conditional random fields," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 415–423, Springer, 2016.
3. O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241, Springer, 2015.
4. Q. Dou, H. Chen, Y. Jin, L. Yu, J. Qin, and P.-A. Heng, "3D deeply supervised network for automatic liver segmentation from CT volumes," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 149–157, Springer, 2016.
5. Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Proceedings of the 26th annual international conference on machine learning*, pp. 41–48, ACM, 2009.
6. M. P. Kumar, B. Packer, and D. Koller, "Self-paced learning for latent variable models," in *Advances in Neural Information Processing Systems*, pp. 1189–1197, 2010.
7. V. Avramova, "Curriculum learning with deep convolutional neural networks," 2015.
8. X. Qi, Z. Liu, J. Shi, H. Zhao, and J. Jia, "Augmented feedback in semantic segmentation under image level supervision," in *European Conference on Computer Vision*, pp. 90–105, Springer, 2016.

9. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.
10. L. Fidon, W. Li, L. C. Garcia-Peraza-Herrera, J. Ekanayake, N. Kitchen, S. Ourselin, and T. Vercauteren, "Generalised wasserstein dice score for imbalanced multi-class segmentation using holistic convolutional networks," *arXiv preprint arXiv:1707.00478*, 2017.
11. C. H. Sudre, W. Li, T. Vercauteren, S. Ourselin, and M. J. Cardoso, "Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations," *arXiv preprint arXiv:1707.03237*, 2017.
12. O. Jimenez-del Toro, H. Müller, M. Krenn, K. Gruenberg, A. A. Taha, M. Winterstein, I. Eggel, A. Foncubierta-Rodríguez, O. Goksel, A. Jakab, *et al.*, "Cloud-based evaluation of anatomical structure segmentation and landmark detection algorithms: Visceral anatomy benchmarks," *IEEE transactions on medical imaging*, vol. 35, no. 11, pp. 2459–2475, 2016.
13. X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 249–256, 2010.
14. P. Goyal, P. Dollár, R. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola, A. Tulloch, Y. Jia, and K. He, "Accurate, large minibatch sgd: Training imagenet in 1 hour," *arXiv preprint arXiv:1706.02677*, 2017.
15. Y. Nesterov, *Introductory lectures on convex optimization: A basic course*, vol. 87. Springer Science & Business Media, 2013.
16. C. Wang and Ö. Smedby, "Multi-organ segmentation using shape model guided local phase analysis," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 149–156, Springer, 2015.
17. G. Vincent, G. Guillard, and M. Bowes, "Fully automatic segmentation of the prostate using active appearance models," *MICCAI Grand Challenge: Prostate MR Image Segmentation*, vol. 2012, 2012.
18. T. Gass, G. Szekely, and O. Goksel, "Multi-atlas segmentation and landmark localization in images with large field of view," in *International MICCAI Workshop on Medical Computer Vision*, pp. 171–180, Springer, 2014.
19. P. Krähenbühl and V. Koltun, "Efficient inference in fully connected crfs with gaussian edge potentials," in *Advances in neural information processing systems*, pp. 109–117, 2011.
20. O. A. J. del Toro and H. Müller, "Hierarchic multi–atlas based segmentation for anatomical structures: Evaluation in the visceral anatomy benchmarks," in *International MICCAI Workshop on Medical Computer Vision*, pp. 189–200, Springer, 2014.
21. R. Kéchichian, S. Valette, M. Sdika, and M. Desvignes, "Automatic 3d multiorgan segmentation via clustering and graph cut using spatial relations and hierarchically-registered atlases," in *International MICCAI Workshop on Medical Computer Vision*, pp. 201–209, Springer, 2014.